

S T E F A N I N T E R N E T G E S C H Ä

D I E N S T A G , 1

Für Symfony einen eigenen Captcha schreiben

Einen eigenen Captcha für das eigene Symfony-Projekt ist relativ einfach, da PHP Funktionen hat um dynamisch Bilder zu erstellen.

Wir haben ein Captcha für unsere Login-Maske erstellt, da Google's reCaptcha zu lange Texte präsentiert und bei zwar seltenen aber doch Gelegenheiten nicht verfügbar war und somit eine Anmeldung auf unsere Webseite unmöglich machte.

Die Lösung besteht aus den folgenden Schritten:

1. Eine kleine Grafik als Hintergrundbild für den Captcha-Text
2. Eine PHP-Klasse, die die Captcha-Grafik produziert
3. Ein Modul "captcha" mit Action und Template
4. Routing für das Captcha-Modul anpassen
5. Eine Formulkasse mit Widget und Validator für das Captcha
6. Ein Validator für das Captcha
7. Das Anzeigen der Captcha-Grafik

1. Hintergrundbild

Oft wir eine etwas "unruhige" Hintergrundgrafik für das Captcha erstellt, um das maschinelle Auslesen des Textes zu erschweren.

Wir haben es hier nicht übertrieben, sondern nur ein par Punkte auf einem weißem Hintergrund platziert.



2. Eine PHP-Klasse für das Erzeugen der Captcha-Grafik

Wir haben uns für eine recht einfache Captche-Grafik entschieden. Es gibt mit PHP auch die Möglichkeit andere Schriftarten zu nutzen sowie den Text zu drehen usw.

Was die einzelne Funktionen in diese Klasse machen können Sie unter <http://php.net> nachschlagen.

Ein kleiner Hinweis noch. Da später der Aufruf unseres Symfony

Ü B E R M



S T E F A N

Ich heiße Stefan Kühn und habe Informatik an der TU Linköping in

Schweden studiert. Seit einigen Jahren arbeite ich als EDV-Berater im Rahmen meiner Firma (Pompadour GmbH). In meine Freizeit segle ich leidenschaftlich gerne mit meinem Katamaran. Diesem Blog möchte ich die Themen Internet, Geschäftsideen und Unternehmertum widmen.

Impressum:

Stefan Kühn
Landgartenstr. 3
97218 Gerbrunn
Tel: 0931-30699709

Diese seite verwendet google analytics.

M E I N P R O F I
A N Z E I G E N

S E I T E N

<http://www.pompadour.info>
<http://www.kaelte-log.de>
<http://www.cafe-karte.de>
<http://www.kirchen-karte.de>
<http://www.firmendienste24.de>
[http://www.unternehmermontag.d](http://www.unternehmermontag.de)
e

Social Networking:

[facebook](#)
[twitter](#)
[xing](#)

L E S E R

Ein kleiner Hinweis noch. Da später der Aufruf unseres Symfony-Moduls eine Grafik liefern soll und kein HTML, ist es wichtig, dass wir den Header mit

```
header ("Content-type: image/png");
```

definieren (siehe unten). Wir sehen hier auch, dass unsere Grafik eine png ist. Es sind aber auch andere Formate möglich. Siehe <http://php.net>.

```
class CaptchaImage {

function generate($CaptchaText) {

$fontsize = 5;

header ("Content-type: image/png");

$image =
ImageCreateFromPNG($_SERVER['DOCUMENT_ROOT'].'./images/captcha.png');
//Backgroundimage

imagecolorallocate ($image, 0, 0, 0);

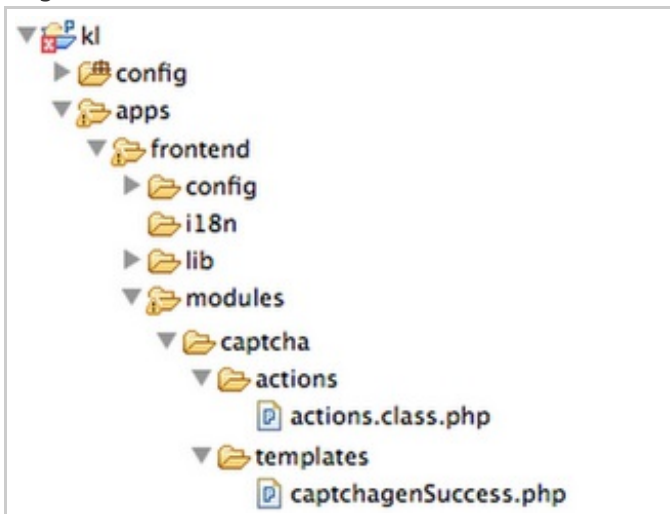
$text_color = ImageColorAllocate ($image, 7, 29, 192);

ImageString ($image, $fontsize, 5, 7, $CaptchaText, $text_color);
ImagePNG ($image);
ImageDestroy($img);
}
}
```

3. Ein Symfony-Modul

Da wir die Captcha-Angelegenheit schön in Symfony-Stil organisiert haben möchten, haben wir ein eigenes Modul dafür erstellt.

Das ist einfach und bringt etwas Struktur in die Anwendung. Das Modul trägt den Namen "captcha" und die Action "captchagen"



B L O G - A

▼ 2011 (3)

▼ Februar (1)

Für Symfony einen eigenen
Captcha schreiben

► Januar (2)

► 2010 (7)

► 2009 (9)

Das Template captchaSuccess.php ist leer, da die Action das Captcha-Bild liefern soll.

Die "captchagen" Action ist nicht sehr umfangreich und besteht hauptsächlich aus dem Generieren des Captche-Textes und das Erzeugen des Bildes. In der Action wir der Captcha-Text in der Session aufgehängt, damit wir den Später gegen Benutzereingabe in unserem Validator prüfen können. Mehr dazu später.

```
class captchaActions extends sfActions
{

    public function executeCaptchagen($request) {
        //Captche-Text generieren

        $captchatext = '';

        $pool1 = 'ABDEFGHIJKLMNOPRSTUVWXYZ';

        for ($i = 1; $i <= 3; $i++) {

            $captchatext .= substr($pool1, rand(0, 22), 1);

        }

        $captchatext .= " ";

        $pool2 = '123456789';

        for ($i = 1; $i <= 3; $i++) {

            $captchatext .= substr($pool2, rand(0, 8), 1);

        }


        //Captche-Text in der Session aufhängen

        $_SESSION['captchatext'] = $captchatext;


        //Captcha-Text mit Hilfe unsere Klasse "CaptchaImage" erstellen

        $tmpCaptchaImage = new CaptchaImage(); $tmpCaptchaImage->generate($captchatext);

    }

}
```

4. Routing anpassen

Damit wir unser Somfony-Modul "captcha" rufen können, passen wir das Routing an.
Das machen wir in der Datei apps/frontend/config/routing.yml

```
captcha:
    url: /captcha
    param: { module: captcha, action: captchagen }
```

5. Widget und Validator für das Captcha

Wir benutzen bereits das sfDoctrineGuardPlugin und haben

daher die darin enthaltene Form-Klasse "sfGuardFormSignin" um unser Captcha erweitert. Selbstverständlich können Sie Ihre eigene Form-Klasse für die (Anmelde-)Maske entsprechend erweitern.

So sieht die erweiterte Klasse aus:

```
class sfGuardFormSignin extends BasesfGuardFormSignin
{
public function configure()
{
//Widgets erstellen und konfigurieren

$this->widgetSchema['username'] = new sfWidgetFormInputText();

$this->widgetSchema['password'] = new
sfWidgetFormInputPassword(array('type' => 'password'));

$this->widgetSchema['remember'] = new sfWidgetFormInputCheckbox();
$this->widgetSchema['captcha'] = new sfWidgetFormInputText();


//Validators erstellen und konfigurieren

$this->validatorSchema['username'] = new sfValidatorString();
$this->validatorSchema['password'] = new sfValidatorString();
$this->validatorSchema['remember'] = new sfValidatorBoolean();
$this->validatorSchema['captcha'] = new sfValidatorCaptcha();


$this->validatorSchema->setPostValidator(new
sfGuardValidatorUser());


$this->widgetSchema->setLabels(array(
'username' => 'Benutzername:',
'password' => 'Passwort:',
'remember' => 'Bitte heute eingeloggt bleiben:',
'captcha' => 'Sicherheitscode:'
));
}
}
```

Wir haben in diese Klasse folgende Erweiterungen gemacht:
Ein Captcha-Widget hinzugefügt:

```
$this->widgetSchema['captcha'] = new sfWidgetFormInputText();
```

Das ist einfach ein normales Textfeld, wo der Anwender den Captcha-Text eintragen kann.

Ein Validator für den Captcha hinzugefügt:

```
$this->validatorSchema['captcha'] = new sfValidatorCaptcha();
```

Dieser Validator, den wir selber geschrieben haben, wird unten erläutert.

6. Validator

Um den, vom Anwender eingegebener, Captcha-Text zu prüfen, schreiben wir in Symfony-Manier ein Validator. Wir haben den "sfValidatorCaptcha" genannt und in der Datei sfValidatorCaptcha.class.php unter lib/form/ abgelegt.

```
class sfValidatorCaptcha extends sfValidatorString
{
    protected function configure($options = array(), $messages =
    array())
    {
        parent::configure($options, $messages);

        $this->setMessage('invalid', 'Der Sicherheitscode stimmt nicht mit
        dem im Bild überein');
    }

    protected function doClean($value)
    {
        parent::doClean($value);

        $clean = (string) $value;

        //Captcha-Text aus der Session holen
        $captchatext = $_SESSION['captchatext'];

        if($captchatext != $clean) {
            throw new sfValidatorError($this, 'invalid', array('value' =>
            $value));
        }

        return $clean;
    }
}
```

In der Methode configure, definieren wir den Hinweis für den Anwender, sollte er den Captcha-Text in der Captcha-Grafik falsch abtippen.

Die Methode doClean vergleicht den vom Anwender eingegebenen Text, mit dem in der Action captchagen generierten Text (den wir aus der Session holen). Rekapitulation: Der Captcha-Text hatten wir ja in der Action "captchagen", wo der Text generiert wird, in der Session aufgehängt, damit der uns später zwecks Validierung zur Verfügung steht.

7. Anzeigen der Captcha-Grafik

Für das Anzeigen der Captcha-Grafik, erweitern wir das entsprechende Template des sfDoctrineGuardPlugins.

Da wir in dem Template nicht das Symfony-Rendering nutzen, sondern ein eigenes Layout mit eine Tabelle erstellt haben, müssen wir hier nur diese Zeilen aufnehmen:

```
<tr>

<td colspan="2">

<?php echo $form['captcha']->renderError() ?>

</td>

</tr>

<tr>

<td>

Sicherheitscode:

</td>

<td>

<?php echo image_tag(url_for('@captcha'), array('raw_name' => true))

?>

</td>

</tr>

<tr>

<td>

<?php echo $form['captcha']->renderLabel() ?>

</td>

<td>

<?php echo $form['captcha']->render() ?>

</td>

</tr>
```

Das interessanteste hier ist diese Zeile:

```
<?php echo image_tag(url_for('@captcha'), array('raw_name' => true))

?>
```

Die Symfony-Hilfsroutine "image_tag" generiert ein img-Tag. In unserem Fall sieht das Ergebnis so aus:

```

```

Der Hilfsroutine-Parameter "raw_name" verhindert, dass Symfony ein ".png" hinten dran an "/captcha" hängt. Auch wenn der Header ein Bild in png-Format vorgibt, wäre das ja nicht korrekt, da hier die Bildquelle (src) eine Symfony-Action ist. Vergleiche hier den url_for-Parameter "@captcha" mit unser Routing-Definition unter Punkt 4 oben. Das hier verwendete "@captcha" ist der Routing-Name "captcha:".

Das fertige Ergebnis sieht so aus:



Ich denke das war's jetzt :-)
Viel Spaß beim Ausprobieren!

PS. Wenn Jemandem veraten könnte, wie man hier Quellcode
formatieren kann, wäre das sehr nett ;-)

E I N G E S T E E A U M I V O G H O N M E
M e t f
L A B S E Y L M S F : O N Y C A P T C H A

S O N N T A G , 3 0

Einen erfolgreichen Unternehmer...

...traf ich dieses Wochenende auf einer Motorradmesse in
Dettelbach bei Würzburg.

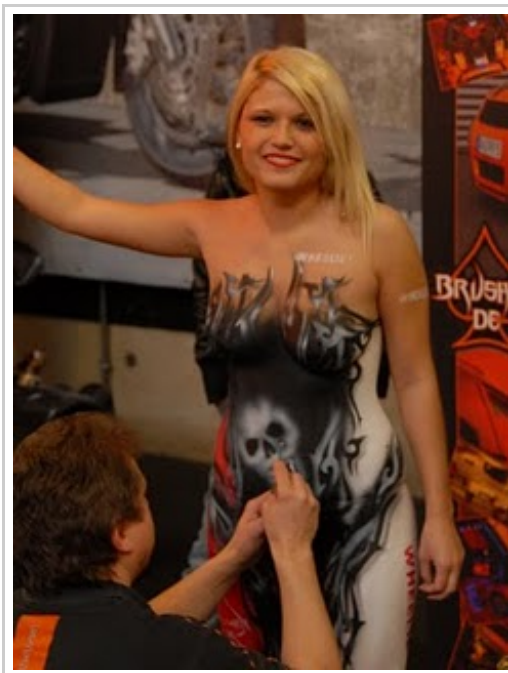
Der Airbrush-Künstler ist unglaublich talentiert und hat sein
Bodypainting-Model in Ca. 20 Minuten bemalt. Das Ergebnis
können Sie unten bestaunen.

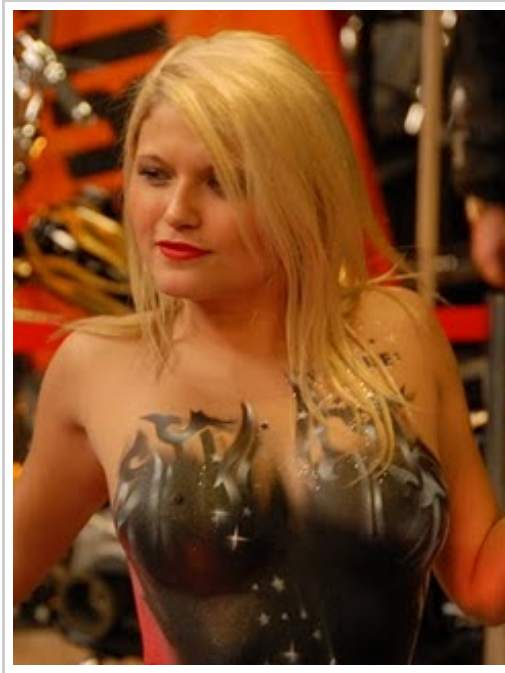
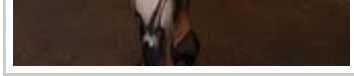
(ein Klick auf das Bild bringt eine Vergrößerung)

Schön ist es auch, wenn man sein Hobby zum Beruf machen
kann.

Sie finden den Künstler unter <http://www.brusher.de>











Ob ich nur wegen den Mädels da war? ;-)
Nein... :-)



E I N G E S T E E A 2 V O U H O N M  E
   

D O N N E R S T A G ,

Bookmarks aus Delicious in XML-Format exportieren

Da Yahoo eventuell den Bookmark-Dienst <http://www.delicious.com/> schließen will, ist es empfehlenswert die Bookmarks zu sichern. Auf der Webseite von Delicious gibt

es ein Export in HTML
(<https://secure.delicious.com/settings/bookmarks/export>).

Wenn Sie aber die Bookmarks in XML exportieren möchten, können Sie diesen Befehl auf der Kommandozeile nutzen:

```
curl https://{username}:{password}@api.del.icio.us/v1/posts/all > bookmarks.xml
```

Voraussetzung ist, dass Sie das Programm curl installiert haben. Auf dem Mac ist es schon drauf ;-)

In dem Befehl oben ersetzen Sie username und password mit Ihren Daten. Die geschweiften Klammern müssen bleiben!

Viel Erfolg!
Und hoffen wir, dass Yahoo sich es anders überlegt!



D I E N S T A G , 9

iPad Kalender-Problem

Seit dem Apple das iCal Kalenderformat (im Oktober?) auf CalDAV umgestellt hat, gibt es probleme mit dem Kalender-Sync auf dem iPad.

Es sollte vielleicht vorsichtshalber erwähnt werden, dass das neue CalDAV-Format eine Option ist. Wenn man sich in MobileMe unter me.com anmeldet und den Kalender anschaut, bekommt man das Angebot auf das neue Format umzustellen.

Es sieht jetzt so aus, als ob die Kalender-Software auf dem iPad mit CalDAV nicht ganz klar kommt. Sprich es synchronisiert überhaupt nicht. Kein einziges Kalenderdatum wird vom MobileMe übernommen. Auf Geräten wie z.B. iPhone die schon das neue Betriebssystem (iOS 4.x) gibt es keine Probleme und das Problem mit dem iPad wird höchstwahrscheinlich jetzt in November gelöst, wenn die version 4.2 für das iPad kommt.

Nachdem ich zig male den MobileMe Account auf dem iPad gelöscht und wieder angelegt hatte, ohne den erhofften Erfolg, fand ich im Netz eine Lösung.






Bevor Sie mit dem Link unten weitermachen, konfigurieren Sie MobileMe wie gewohnt auf Ihrem iPad (unter Account hinzufügen wählen Sie "MobileMe" aus). Sie deaktivieren aber hier "Kalender". Somit werden Mail, Kontakte und Lesezeichen über diesem Account synchronisiert.

Jetzt...lesen Sie hier weiter. Hier wird beschrieben, wie man ein CalDAV Account anlegt.

Und plötzlich erscheinen wieder Ihre Termine in dem neuen Kalender-Format von MobileMe :-)

Viel Erfolg!

<http://help.apple.com/mobileme/interface/index.html?lang=en#mmcc733205>

E I N G E S T E E A U T O 3 V O Ü H O N M  E
    

D I E N S T A G , 2

Time Machine: indizieren verhindern

Ist es dir als Mac OSX Anwender aufgefallen, dass die Time Machine inzwischen jedes mal das Backup Volume (Time Capsule) indiziert, bevor der Backup anläuft?

Höchstst wahrscheinlich ist das Verhalten mit einem System-Update dazu gekommen.

Das Indizieren soll es ermöglichen mit Spotlight das Backup-Volume zu durchsuchen.


Mir stört's und ich brauch's nicht. Netter weise lässt es sich leicht abstellen.

In den Systemeinstellungen unter "Spotlight", Lasche "Privatsphäre", wird einfach das Volume Time Machine Backups hinzugefüht. Ein Klick auf das + bringt die Auswahlliste.

Vorausgesetzt das Volume ist gerade gemountet.

So sieht es aus:



E I N G E S T E E A U T O 2 V O Ü H O N M  E
    

"PomPay" auf dem Forum Kiedrich Gründermarkt am 26.06.2010

Gestern nahm ich an dem Forum Kiedrich Gründermarkt (<http://www.forum-kiedrich.de>) in Wiesbaden als Gründer teil. Dabei hatte ich zwei meiner Gründungspartner Stefan Keller und Peter Wiesel.

Zunächst muss ich ein spontanes Lob an den Organisatoren aussprechen. Ich denke das Forum Kiedrich ist einmalig in Deutschland. Einfach phantastisch!

Es ist den Gründern und Betreibern von Forum Kiedrich gelungen eine unglaublich kreative und familiäre Atmosphäre zu schaffen. Hier treffen sich sowohl Gründer und sehr erfahrene Unternehmer auf einer Ebene obwohl die „Abstände“ wohl kaum größer sein könnten. Ich hatte selber die Gelegenheit den Unternehmensberater Herr Klaus Christian Plönzke und Frau Ingeborg Mehren-Hitchcock persönlich zu sprechen. Ich sage es ganz offen. Ich habe es als eine Ehre empfunden.

Herr Plönzke ist nicht nur Initiator des am 6. April 1997 gegründeten Gründerinitiative "Forum Kiedrich" er hat auch am 29. September 2003 das Bundesverdienstkreuz 1. Klasse des Verdienstordens der Bundesrepublik Deutschland für seine Leistungen beim Aufbau der deutschen IT-Branche und sein Engagement für junge Unternehmensgründer erhalten.

Frau Ingeborg Mehren-Hitchcock ist nicht weniger faszinierend. Nicht nur den berühmten Hitchcock-Orangensaft brachte sie nach Deutschland, sondern als kluge Geschäftsfrau in den Sechzigerjahren auch zahllose andere Produkte, die den American Way of Life zunächst nur in der Stadt Wiesbaden und dann in der ganzen Bundesrepublik populär machten. In Mähren geboren und in Salzburg aufgewachsen, studierte Ingeborg Mehren-Hitchcock im texanischen Austin, wo sie bald für eine Stahlfirma arbeitete, in der sie ihren späteren Ehemann, den Ingenieur Bill Hitchcock (einen Cousin des Regisseurs Alfred Hitchcock) kennenlernte. In ihrer zweiten Ehe mit George Mehren, dem stellvertretenden Landwirtschaftsminister der Regierung Lyndon B. Johnson, bereiste sie zahlreiche Länder der Dritten Welt und widmete sich zwischen ihren Wahlheimaten Texas und Wiesbaden dem Familienleben sowie dem Fundraising für wohltätige Zwecke.


Mit Frau Mehren-Hitchcock hätte ich noch stunden lang reden können und immer noch das Gefühl haben, es war zu kurz!

Es waren aber nicht nur die „Großen“, die das Forum Kiedrich mit Leben füllten. Es waren auch die Vielen anderen sehr guten

Kontakte zu anderen Gründern und Kapitalgebern.

Ich kann wirklich jeden Gründer empfehlen, sich für das Forum Kiderich zu bewerben. Das ist eine einmalige Erfahrung die zu dem, durch eine mögliche Mitgliedschaft, lange erhalten bleibt.

Vielen Dank, dass wir mit unserem Projekt „PomPay“ teilnehmen dürfen!

E I N G E S T E E A M 2 W O Ü H O N M  E



M I T T W O C H , 1

Abgabeparty der Phase 3 des Businessplan-Wettbewerbs Netzwerk Nordbayern

Gestern (15.06.2010) besuchte ich das Seminar „Grundlagen der Unternehmensfinanzierung“, organisiert vom Netzwerk Nordbayern (<http://www.netzwerk-nordbayern.de>).

Es gab eine Reihe sehr interessante Vorträge von folgenden Personen und Organisationen:

Begrüßung und Hintergrundinfo von Frau Müller, Netzwerk Nordbayern (<http://www.netzwerk-nordbayern.de>)

Info zu Venture Capital von Herr Raab, S-Refit in Regensburg. (<http://www.s-refit.de>)

Info zu Frühphasenfinanzierung von Herrn Redl, Bayern Kapital in Landshut (<http://www.bayernkapital.de>)

Info zu öffentliche Förderung, Herr Leitenmayer, LfA Förderbank Bayern. (<http://www.lfa.de>)

Info zu Bankenfinanzierung, Herr Schilling, Sparkasse Nürnberg. (<http://www.sparkasse-nuernberg.de>)

Info zu Technologieförderung in Bayern, Herr Lorenz, LGA Nürnberg (<http://www.lga.de>)

Alles in allem war es eine sehr gut organisierte Veranstaltung.

Im Anschluss fand das Abgabeparty der Phase 3 des Businessplan-Wettbewerbs, ebenfalls vom Netzwerk Nordbayern organisiert.

Dieser Event wurde mit drei Vorträgen eingeleitet:

Her Roman Huber, der Geschäftsführer der Bayern Kapital,

erzählte auf eine sowohl informative als auch amüsante Art und Weise über die Tätigkeiten von Bayern Kapital. Ein Vortrag der, an diesem Abend, besonders gut ankam.

Herr Peter Pauli, Geschäftsführer der BayBG (<http://www.baybg.de>) präsentierte ausführlich und informativ über deren Tätigkeiten.

Zuletzt erzählte Herr Dr. Huber über die eigenen Erfahrungen als Gründer von LipoFIT (<http://www.lipofit.de>). Sehr amüsant und aufschlussreich!

Der Abend ließen alle mit einem leckerem Buffet und jede Menge Gelegenheiten zum Networking ausklingen. Alles vom Netzwerk Nordbayern perfekt organisiert!

Ich kann wirklich jedem Gründer Empfehlen, mit Netzwerk Nordbayern Kontakt auf zu nehmen. Es ist gut investierte Zeit!



[Startseite](#)

[Ältere Posts](#)

Abonnieren [Posts \(Atom\)](#)